IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
Before the Board of Patent Appeals and Interferences

In re Patent Application of
NEVILL et al.
Serial No. 09/887,560
Filed: June 25, 2001
Title: STORING STACK OPERANDS IN REGISTERS

Atty Dkt. 550-243
C# M#
TC/A.U.: 2183
Examiner: B. O'Brien
Date: June 21, 2005

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

☐ **Correspondence Address Indication Form Attached.**

☐ **NOTICE OF APPEAL**
Applicant hereby **appeals** to the Board of Patent Appeals and Interferences from the last decision of the Examiner twice/finally rejecting applicant's claim(s). $500.00 (1401)/$250.00 (2401) $

☒ An appeal **BRIEF** is attached in the pending appeal of the above-identified application $500.00 (1402)/$250.00 (2402) $ 500.00

☐ Credit for fees paid in prior appeal without decision on merits -$ ( )

☐ A reply brief is attached. **(no fee)**

☐ Petition is hereby made to extend the current due date so as to cover the filing date of this paper and attachment(s)
One Month Extension $120.00 (1251)/$60.00 (2251)
Two Month Extensions $450.00 (1252)/$225.00 (2252)
Three Month Extensions $1020.00 (1253/$510.00 (2253)
Four Month Extensions $1590.00 (1254/$795.00 (2254) $

☐ "Small entity" statement attached.

Less month extension previously paid on -$( )

TOTAL FEE ENCLOSED $ 500.00

Any future submission requiring an extension of time is hereby stated to include a petition for such time extension. The Commissioner is hereby authorized to charge any deficiency, or credit any overpayment, in the fee(s) filed, or asserted to be filed, or which should have been filed herewith (or with any paper hereafter filed in this application by this firm) to our **Account No. 14-1140.** A duplicate copy of this sheet is attached.

901 North Glebe Road, 11th Floor
Arlington, Virginia 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100
SCS:kmm

NIXON & VANDERHYE P.C.
By Atty: Stanley C. Spooner, Reg. No. 27,393

Signature:

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Patent Application of	Confirmation No.: 7549
NEVILL et al.	Atty. Ref.: 550-243
Serial No. 09/887,560	Group: 2183
Filed: June 25, 2001	Examiner: A. Li

For: STORING STACK OPERANDS IN REGISTERS

* *

APPEAL BRIEF

On Appeal From Group Art Unit 2183

Stanley C. Spooner
NIXON & VANDERHYE P.C.
8th Floor, 1100 North Glebe Road
Arlington, Virginia 22201-4714
(703) 816-4028
Attorney for Appellant

TABLE OF CONTENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Patent Application of

NEVILL et al. Atty. Ref.: 550-243

Serial No. 09/887,560 Group: 2183

Filed: June 25, 2001 Examiner: B. O'Brien

For: STORING STACK OPERANDS IN REGISTERS

* *

June 21, 2005

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Sir:

I. REAL PARTY IN INTEREST

The real party in interest in the above-identified appeal is ARM LIMITED by virtue of an Assignment of rights from the inventors to ARM LIMITED recorded October 11, 2001 at Reel 12249, Frame 0290.

II. RELATED APPEALS AND INTERFERENCES

There are believed to be no related appeals, interferences or judicial proceedings with respect to the present application and appeal.

III. STATUS OF CLAIMS

Claims 1-9 and 12-17 stand rejected in the outstanding Final Rejection. The Examiner contends that claims 1-9 and 12-17 are anticipated by Patel (U.S. Patent 6,332,215).

IV. STATUS OF AMENDMENTS

No further response has been submitted with respect to the Final Official Action in this application.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

The present invention relates to the field of data processing systems and particularly data processing systems which support multiple instruction sets.

One example discussed in the background of the present application is the Thumb enabled processors produced by the Assignee, which may execute both 16-bit Thumb instructions and 32-bit ARM instructions. Thus, the data processing system supports two instruction sets. Both of the instructions execute operations, such as mathematical manipulations, loads, stores, etc., based upon operands stored within registers of the processor core specified by register fields within the instructions.

Another class of instruction sets are those which use a stack approach to storing and manipulating the operands upon which they act. The stack may store a

sequence of operand values. One example of a stack based instruction set is the Java Virtual Machine instruction set specified by Sun Microsystems Inc.

Appellants have developed a new way for a data processing apparatus to operate with multiple instruction sets, where a first instruction set is based upon execution of operations on register operands held in a plurality of registers and a second instruction set specifies operations to be executed upon stack operands held in a stack. Appellants have developed an instruction translator which can translate the instructions of the second instruction set (operands held in a stack) into instructions in accordance with the first instruction set (register operands held in said register).

One significant aspect of the claimed invention is that the "instruction translator", used to translate instructions of the second instruction set into output signals which correspond to instructions of the first instruction set, utilizes a plurality of "instruction templates." These templates translate instructions from the second instruction set to instructions from the first instruction set. Where the second instruction set includes one or more stack operands, there is an instruction template comprising one or more instructions from the first instruction set in which register operands are mapped to the stack operands in dependence upon a currently adopted mapping state of the instruction translator.

In other words, the instruction translator allows a data processing system utilizing register operands held in a plurality of registers in accordance with a first

instruction set to utilize a second instruction set which is based upon operations executed upon stack operands held in a stack through the use of "instruction templates."

Thus, essential to the operation of the present invention which allows multiple instruction sets to be carried in a single data processing system, there is provided an instruction translator 42 which utilizes a plurality of instruction templates for translating instructions from the second non-native instruction set into native instructions. This is discussed in Appellants' specification, beginning at page 9, line 3. Thus, the flexibility benefit recognized by Appellants' invention is a result of the use of templates having associated mappings of register operands to stack operands, as clearly set out in Appellants' claim 1 clause (vi).

Thus, Appellants' independent apparatus claim 1 and independent method claim 16 are characterized by having:

"**a processor core having a register bank containing a plurality of registers** and being operable to execute operations upon register operands held in said registers as specified within instructions of a first instruction set;" and

"**an instruction translator operable to translate instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set** . . . wherein said instruction translator uses **a plurality of instruction templates** for translating instructions from said second instruction set to instructions from said first instruction set." (emphasis added).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1-9 and 12-17 stand rejected under 35 USC §102(e) as being anticipated by Patel (U.S. Patent 6,332,215).

VII. ARGUMENT

Appellants' arguments include the fact that the burden is on the Examiner to demonstrate where the single cited reference teaches each of the structures and/or method steps recited in independent claims 1 and 16.

The Court of Appeals for the Federal Circuit has noted in the case of *Lindemann Maschinenfabrik GMBH v. American Hoist & Derrick*, 221 USPQ 481, 485 (Fed. Cir. 1984) that "[a]nticipation requires the presence in a single prior art reference disclosure of each and every element of the claimed invention, arranged as in the claim."

A. Claims 1 and 16 require "an instruction translator" which "uses a plurality of instruction templates"

Claim 1 recites an instruction translator which "uses a plurality of instruction templates for translating instructions from said second instruction set to instructions from said first instruction set" and where "an instruction from said second instruction set including one or more stack operands has an instruction template comprising one or more instructions from said first instruction set in

which register operands are mapped to said stack operands in dependence upon a currently adopted mapping state of said instruction translator."

Independent claim 16 requires a "translating instructions" step using "a plurality of instruction templates" including "at least one instruction from said first instruction set in which register operands are mapped to said stack operands in dependence upon a currently adopted mapping state of said translating step."

In order to anticipate independent claims 1 and 16, it is necessary for the Examiner to demonstrate that the Patel prior art reference teaches the same invention claimed, i.e., one which uses "a plurality of instruction templates" as set out in the claims.

B. The Examiner does not support his allegation that Patel teaches instruction templates with citation to any portion of Patel

The Examiner suggests that Patel teaches an instruction translator which "uses a plurality of instruction templates for translating instructions from said second instruction set to instructions from said first instruction set" (page 3, subsection f. of the Final Rejection). However, Patel does not specifically disclose the use of the template associated with the non-native instruction that uses a mapping that depends upon the currently adopted mapping state of an instruction translator as in Appellants' independent claims.

Patel discloses in column 5, lines 32-39 (the portion relied upon by the Examiner in the Final Rejection), that a "state machine 74 provides an address that indicates the location of a desired native instruction in a look-up table 78. Counters are maintained to keep a count of how many entries have been placed on the operand stack, as well as to keep track of the operand stack. In a preferred embodiment of Patel, the output of the look-up table 78 is augmented with indications of the registers to be operated on at line 80." However, there is no discussion of the use of a single instruction template, let alone "a plurality."

As a result, the Examiner has failed to support the rejection under 35 USC §102.

C. Patel teaches away from the invention of Claims 1 and 16.

The Examiner does not point to any language which suggests that Patel teaches the use of templates having associated mappings of register operands to stack operands as specified in claims 1 and 16. In point of fact, Patel actually teaches away from the structure of claim 1 and method of claim 16 because Patel teaches that counters are required in addition to the look-up table 78.

Patel specifically requires that separate counters are used to keep track of how many empties have been placed on the operand stack and also to keep track of the top of the operand stack. Patel also teaches that it is required to augment the output of the look-up table 78 with indications of the registers to be operated on.

Thus, the mapping between register operands and stack operands in the Patel system is not in fact a function performed by the look-up table 78 (and thus it cannot be an instruction template).

Patel instead requires a separate system of keeping track of the number of operands in the operand stack and the top stack operands using counters. Thus, Patel teaches a different way of operating from that set out in Appellants' independent claims 1 and 16, i.e., using an instruction translator for translating instructions from the second instruction set to the first instruction set by way of templates having associated mappings of register operands to stack operands.

Because Patel teaches away from the subject matter of Appellants' claimed invention, it cannot anticipate or render obvious the subject matter of independent claims 1 and 16 or claims dependent thereon. As a result, there is no support for the single rejection of claims 1-9 and 12-17 under 35 USC §102 as being anticipated by Patel.

VIII. CONCLUSION

Appellants' independent claims 1 and 16 specify that the claimed invention includes an instruction translator which "uses a plurality of instruction templates for translating instructions from said second instruction set to instructions from said first instruction set." This specific teaching is completely missing from the Patel reference, and the Final Rejection does not address how or where such

teaching is contained in the Patel reference. Where a prior art reference does not teach every structure and/or method step, as a matter of law, it cannot anticipate a claim. As a result, there is no support for the Examiner's anticipation rejection of claims 1-9 and 12-17 and any further rejection is respectfully traversed.

Thus, and in view of the above, the rejection of claims 1-9 and 12-17 under 35 USC §102 is clearly in error and reversal thereof by this Honorable Board is respectfully requested.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: _____

Stanley C. Spooner
Reg. No. 27,393

SCS:kmm
Enclosure

IX. CLAIMS APPENDIX

1. An apparatus for processing data, said apparatus comprising:

(i) a processor core having a register bank containing a plurality of registers and being operable to execute operations upon register operands held in said registers as specified within instructions of a first instruction set; and

(ii) an instruction translator operable to translate instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set, instructions of said second instruction set specifying operations to be executed upon stack operands held in a stack; wherein

(iii) said instruction translator is operable to allocate a set of registers within said register bank to hold stack operands from a portion of said stack;

(iv) said instruction translator has a plurality of mapping states in which different registers within said set of registers hold respective stack operands from different positions within said portion of said stack;

(v) said instruction translator is operable to change between said plurality of mapping states in dependence upon operations that add or remove stack operands held within said set of registers; and

(vi) wherein said instruction translator uses a plurality of instruction templates for translating instructions from said second instruction set to instructions from said first instruction set and wherein an instruction from said second instruction set including one or more stack operands has an instruction template comprising one or more instructions

from said first instruction set in which register operands are mapped to said stack operands in dependence upon a currently adopted mapping state of said instruction translator.

2. An apparatus as claimed in claim 1, wherein said translator output signals include signals forming an instruction of said first instruction set.

3. An apparatus as claimed in claim 1, wherein said translator output signals include control signals that control operation of said processor core and said control signals match control signals produced on decoding instructions of said first instruction set.

4. An apparatus as claimed in claim 1, wherein said translator output signals include control signals that control operation of said processor core and specify parameters not specified by control signals produced on decoding instructions of said first instruction set.

5. An apparatus as claimed in claim 1, wherein said instruction translator provides mapping states such that stack operands are added to or removed from said set of registers without moving stack operands between registers within said set of registers.

6. An apparatus as claimed in claim 1, wherein said set of registers are operable to hold stack operands from a top portion of said stack including a top of stack operand from a top position within said stack.

7. An apparatus as claimed in claim 1, wherein said stack further comprises a plurality of addressable memory locations holding stack operands.

8. An apparatus as claimed in claim 7, wherein stack operands overflow from said set of registers into said plurality of addressable memory locations.

9. An apparatus as claimed in claim 7, wherein stack operands held within said plurality of addressable memory locations are loaded into said set of registers prior to use.

12. An apparatus as claimed in claim 1, wherein said instruction translator comprises one or more of:

(i) hardware translation logic;

(ii) instruction interpreting program code controlling a computer apparatus;

(iii) instruction compiling program code controlling a computer apparatus; and

(iv) hardware compiling logic.

13. An apparatus as claimed in claim 1, wherein said instruction translator includes a first plurality of state bits indicative of a number of stack operands held within said set of registers.

14. An apparatus as claimed in claim 13, wherein said instruction translator includes a second plurality of state bits indicative of which register within said set of registers holds said top of stack operand.

15. An apparatus as claimed in claim 1, wherein said second instruction set is a Java Virtual Machine instruction set.

16. A method of processing data using a processor core having a register bank containing a plurality of registers and being operable to execute operations upon register operands held in said registers as specified within instructions of a first instruction set, said method comprising the steps of:

(i) translating instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set using a plurality of instruction templates, instructions of said second instruction set specifying operations to be executed upon stack operands held in a stack;

(ii) allocating a set of registers within said register bank to hold stack operands from a portion of said stack;

(iii) adopting one of a plurality of mapping states in which different registers within said set of registers hold respective stack operands from different positions within said portion of said stack wherein an instruction from said second instruction set including at least one stack operand has an instruction template comprising at least one instruction from said first instruction set in which register operands are mapped to said stack operands in dependence upon a currently adopted mapping state of said translating step; and

(iv) changing between said plurality of mapping states in dependence upon operations that add or remove stack operands held within said set of registers.

17. A computer program product holding a computer readable medium including computer readable instructions that when executed perform the method of claim 16.